The MagPi magazine is published by Raspberry Pi (Trading) Ltd., Mount Pleasant House, Cambridge, CB3 0RN. ISSN: 2051-9982.

Prova altri metodi di making con #MonthOfMaking 2021 e The MagPi!

È di nuovo quel periodo dell'anno! Marzo, da queste parti, è il #MonthOfMaking, il nostro vento di making della community che dura un mese.

Ci piace cercare e mostrare i fantastici progetti che emergono nelle community più ampie di Raspberry Pi e maker in senso più ampio – e per più ampio, intendiamo davvero più ampio. Nella rivista parliamo spesso di progetti basati sull'elettronica – siamo una rivista di informatica, dopotutto. Tuttavia, il mondo maker comprende molto di più: artigianato, cucito, lavoro a maglia, cottura al forno, scultura, fotografia, cosplay, modellismo e molto altro ancora.

Puoi fare tutto ciò che vuoi per il #MonthOfMaking, nel frattempo in questo numero, vogliamo evidenziare altri modi in cui si può fare making ed essere creativi. Costruiamo!

Come partecipare

Le regole di #MonthOfMaking sono semplici

Lavora su un progetto, nuovo o vecchio

Scatta foto dei tuoi progressi e del progetto completo

Condividilo su Twitter e/o Instagram con una descrizione

Accertati di includere l'hashtag #MonthOfMaking

RISORSE #MONTHOFMAKING

The MagPi 79

Abbiamo introdotto #MonthOfMaking nel numero 79 di The MagPi con undici progetti che potresti provare

bit.ly/MagPi79It

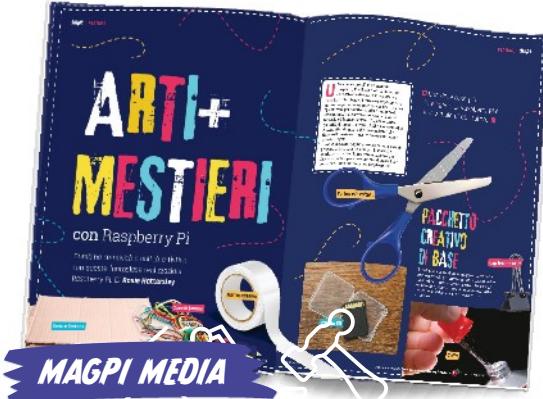
The MagPi 91

In The MagPi numero 91, ti abbiamo mostrato come documentare e condividere i tuoi progetti con la community

bit.ly/MagPi91It

HackSpace Magazine

La nostra pubblicazione sorella, HackSpace magazine, si sta unendo al divertimento di #MonthOfMaking quest'anno. Il suo catalogo è un tesoro di incredibili idee di progetto e ispirazione

hsmag.cc

Hashtag #MonthOfMaking

Guarda cosa stanno facendo gli altri! Lasciati ispirare e offrir aiuto se puoi, quando ti viene chiesto

Making artigianale

Cucito, lavoro a maglia, uncinetto, tessitura, ecc. sono tra le più antiche forme di making. Ecco come farli evolvere con la tecnologia

Di solito, la cosa più simile a una sorta di progetto di cucito che riusciamo a fare con Raspberry Pi sono i dispositivi indossabili. A modo loro sono forti, e possono essere sicuramente parte di questo argomento, ma non facciamo molto per combinare veramente l'artigianato del filato e quello dell'elettronica. Ecco alcuni modi per farlo.

ARTI+ MESTIERI

con Raspberry Pi

PACCHETTO CREATIVO DI BASE

MAGPI MEDIA

Di recente abbiamo realizzato un grande speciale su Arti e Mestieri con Raspberry Pi e puoi trovarlo nel numero 101 di The MagPi (**bit.ly/MagPi101It**) per ulteriori suggerimenti e ispirazione.

Iniziare con gli indossabili

Questo è un ottimo punto di partenza se desideri imparare un po' di più sul cucito e sulla programmazione con filo conduttivo. Utilizza Adafruit FLORA; tuttavia, se vuoi provare a modificarlo per un Raspberry Pi Pico, è più che fattibile.

Cucire è qualcosa che richiede un po' di pratica, ma è un'abilità davvero utile se vuoi modificare o anche creare i tuoi vestiti! L'aggiunta di dispositivi indossabili in questi capi può essere molto divertente ed è anche molto utile quando arriva il tempo dei cosplay.

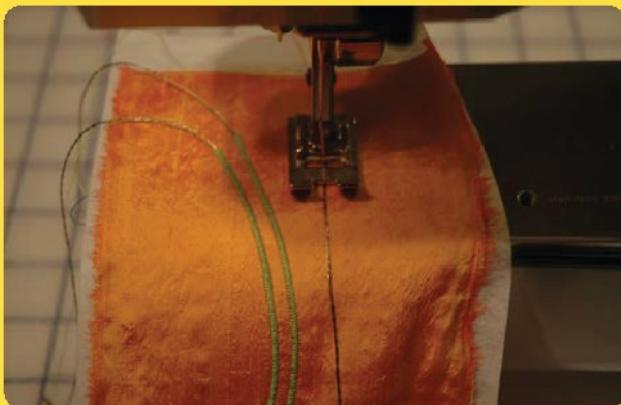
magpi.cc/wearables

ANDIAMO OLTRE...

Social Media senza internet

Questo progetto è più un esempio d'arte: mostra l'applicazione estrema della tecnologia indossabile con pulsanti e schermi dappertutto su queste tute. un "cinque" vale come un "mi piace" in questo social offline sperimentale. Tuttavia, puoi usare alcune delle idee per diversi tipi di costumi - anche per preparare Halloween con molti mesi di anticipo.

magpi.cc/socialwear

Macchina da ricamo CNC

Questa è una bella miscela di tecnologia. Molte moderne macchine da ricamo ti consentono di caricare un modello tramite computer in modo che il risultato finale sia molto più preciso e veloce, se necessita. Questo progetto è incentrato sulla modifica di una normale macchina da cucire che si connette a un Raspberry Pi e ad alcuni motori per fare il duro lavoro. È anche molto più economico, se hai le capacità, i materiali e la pazienza. Questo è un mix di modifiche hardware e software abbastanza avanzato. Se hai voglia di fare qualche altro tipo di riciclo/upcycle, consulta qui gli altri progetti.

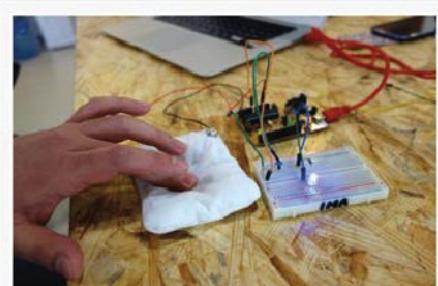
magpi.cc/cncembroid

SUGGERIMENTI DI RICAMO ALTERNATIVO...

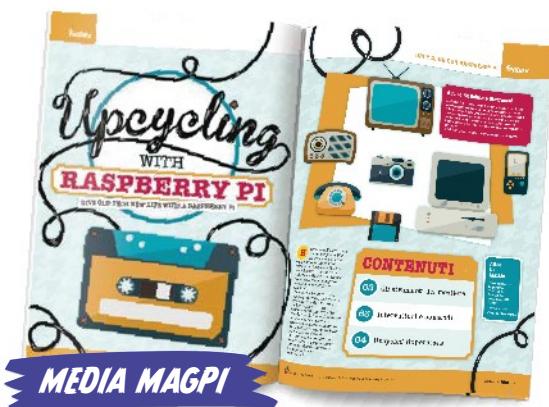
Ricamo con filo conduttivo

Non sei sicuro se vuoi cucire a mano il filo conduttivo in qualche tessuto? Vuoi provare a nasconderlo? Questo geniale metodo di fissaggio del filo conduttivo consiste nell'avvolgerlo al filo tessile e gli consente di essere in qualche modo isolato! Con un po' di pratica e mano ferma, potresti farlo anche tu con una normale macchina da cucire!

magpi.cc/embroidery

PROGETTI TECH INDOSSABILI

I nostri amici di HackSpace Magazine hanno lavorato con il sorprendente Sophy Wong per creare un fantastico libro indossabile, completo di idee incredibili per fondere insieme tecnologia e moda. Dai un'occhiata a **magpi.cc/wearableprojects**.

Pulsante morbido

Questo progetto è un po' più di un modo fai-da-te di lavorare con il tessuto e l'elettronica. Crea un pulsante morbido e sicuro che può essere utilizzato in una varietà di indossabili o altre applicazioni. È anche abbastanza camuffato, per un pulsante. Ci sono anche alcuni buoni consigli su come usare fili conduttivi in una macchina da cucire, come abbassare la tensione sulla macchina. Questo progetto è basato su Arduino, ma è perfetto per Raspberry Pi Pico.

Riciclo e “upcycling” delle cose

Usa tecnologia vecchia e casalinghi indesiderati e regalagli una nuova vita

Fare making può diventare costoso, ma c'è così tanto che puoi fare con la vecchia tecnologia o una scatola di cartone con un po’ di ingegno e di olio di gomito. Ecco alcuni grandi progetti a cui potresti ispirarti per trasformare la tua scatola Amazon in un robot rasa erba.

Ci piace parlare regolarmente con Martin Mander dei suoi progetti di upcycling, quindi abbiamo deciso di scegliere le sue idee per alcuni ottimi consigli di upcycling. Dai un'occhiata al numero 70 di *The MagPi*. **bit.ly/MagPi70It.**

Riciclare

Lampadario con NeoPixel riciclato

Realizzato con bottiglie riciclate per il massimo look fai-da-te, questo lampadario alimentato da NeoPixel è meraviglioso. Utilizza anche del legno riciclato per parte della costruzione. La miscela di bottiglie forse non è del tutto la nostra preferita, ma ha comunque un bell’aspetto. Questo progetto utilizza le schede Adafruit anziché Raspberry Pi, ma è facile sostituirle in modo da poter continuare a utilizzare quello che hai, invece di acquistare qualcosa di nuovo.

magpi.cc/chandelier

Braccio Robotico Riciclato

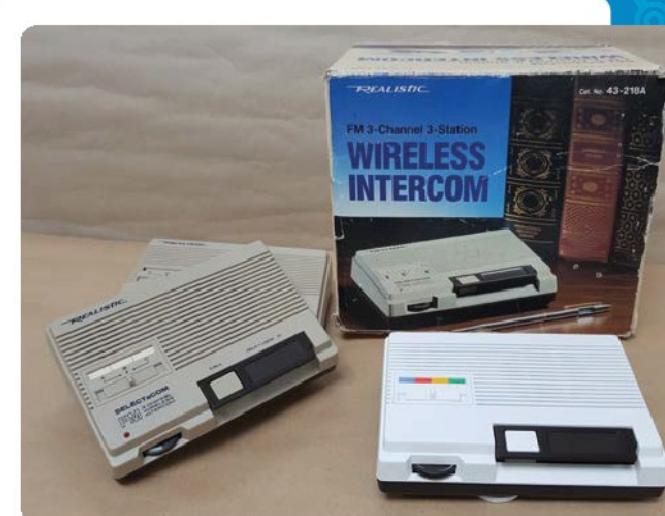
Un robot realizzato con parti riciclate e/o rottami, sembra molto simile a un personaggio di un film di fantascienza. Tuttavia, questa versione è molto reale! Ci ricorda molto il kit robot MeArm e probabilmente potresti usare parti di esso per rendere questo robot più grande. Ancora una volta, è tutto alimentato da Arduino, ma controllare motori e servi con Raspberry Pi è incredibilmente facile utilizzando vari HAT per motori o robot. Anche l'abbinamento di un controller di gioco sarà un po’ più semplice. Il contrappeso fatto con la patata è universale, però.

magpi.cc/recyclearm

"Upcycling"

Game Boy Zero

Infilare un Raspberry Pi Zero all'interno di un classico controller di gioco o di un palmare - o qualsiasi vecchia tecnologia, in tema - rappresenta una misura perfetta per i piccoli dispositivi. Sebbene ci siano molti case in stile Game Boy e altri che puoi stampare in 3D (e che funzioneranno molto bene), questa versione utilizza un vero Game Boy originale, anche se con alcune modifiche esterne, come i pulsanti extra presi da un controller NES. Una cartuccia è stata anche trasformata in un adattatore per schede microSD gigante, così le puoi cambiare facilmente.

magpi.cc/gbzero

Caffettiera - Serra

È un upcycle o un riciclo? Probabilmente è più vicino al riciclo in quanto non funziona come originariamente previsto, ma utilizza la vecchia tecnologia in un modo nuovo. Ad ogni modo, è molto interessante, poiché è stata trasformata in un abbeveratoio automatizzato per fioriera che utilizza le funzioni originali della macchina.

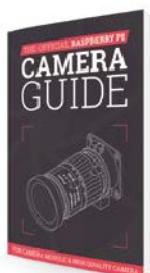
Il sistema può rilevare quando il terreno è asciutto, anche se attualmente si limita a irrigare con un timer. Il pulsante di erogazione sul telaio originale può essere utilizzato per erogare l'acqua in anticipo sul programma, che è qualcosa che vorremmo vedere di più, nei progetti automatizzati.

magpi.cc/greencoffee

Intercomunicante Google Pi

Sarebbe molto facile riempire le pagine di progetti di upcycling del sempre inventivo e creativo Martin Mander. Questo Google Pi Intercom rimane uno dei nostri preferiti grazie alla sua miscela di vecchia tecnologia (un intercomunicante da ufficio anni Ottanta) e nuova tecnologia (l'assistente software Google Voice). Lo ha fatto utilizzando il kit Google AIY Voice distribuito gratuitamente con il numero 57 di The MagPi, ma non è necessario per farlo funzionare, al giorno d'oggi.

Puoi trovare tecnologia di seconda mano e non funzionante praticamente su tutti i siti di aste o di vendita online - assicurati di aver scaricato completamente tutti i condensatori prima di giocare con essi.

magpi.cc/piintercom

Fotografia con Raspberry Pi

Scatta foto con Raspberry Pi, e fai perfino animazioni

La fotografia è un'arte e una scienza. Modificare le impostazioni di una fotocamera per una composizione perfetta va di pari passo con la creazione di foto eccellenti. Puoi fare così tanto con un Raspberry Pi e una fotocamera, indipendentemente dal fatto che utilizzi un Camera Module / High Quality Camera ufficiale, una webcam USB, o una DSLR modificata.

MEDIA MAGPI

Impara tutto ciò di cui hai bisogno di sapere sull'uso della fotocamera ufficiale Raspberry Pi con *The Official Raspberry Pi Camera Guide*: **magpi.cc/cameraguide**.

REGOLA DEI TERZI

Un modo rapido e semplice per inquadrare la tua foto è pensare ad essa in terzi – tipo una griglia 3×3 che suddivide la foto, come nell'immagine sopra. Il soggetto della foto, come l'orizzonte o un albero, dovrebbero trovarsi all'incirca su una delle linee o su un'intersezione delle stesse, per avere un maggior risalto.

High Quality Camera hack del case

Desideri un modo semplice per portare in giro una Raspberry Pi HQ Camera e usarla come una fotocamera tradizionale inquadra e scatta? Richard Hayler della Fondazione Raspberry Pi e la sua famiglia ci mostrano questa ottima modifica che utilizza un case Raspberry Pi ufficiale. È una modifica permanente che comprende in alcuni fori, quindi sii certo che non ti dispiaccia tagliare una custodia di ricambio – e fallo con la supervisione di un adulto se ne hai bisogno!

bit.ly/MagPi94It

Fai una camera 3D

Sebbene tu possa fare alcune fantastiche modifiche digitali a una singola foto per renderla più tridimensionale, il modo migliore e più originale è scattare due foto nello stesso tempo, utilizzando lenti affiancate. L'assiduo collaboratore di *The MagPi*, PJ Evans, mostra come utilizzare due High Quality Camera e due schede Raspberry Pi Zero collegate insieme.

Dimostra anche come visualizzare le fotografie – qualcosa che tornerà molto utile per impressionare gli amici.

magpi.cc/3dcamera

FRAME AL SECONDO

Quando crei un'animazione, idealmente vuoi mirare ad avere 24 frame al secondo, che è il frame rate tradizionale del cinema. Questo si chiama "animating on ones". Può richiedere molto lavoro però, quindi alcuni animatori saltano un fotogramma o tre e così verrà risparmiato tempo.

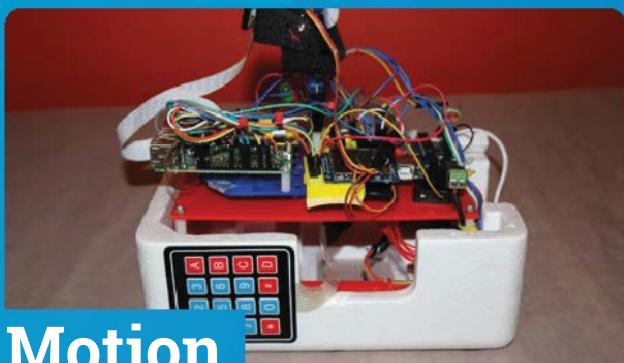
Frame al secondo	Nuova foto ogni	Nome
24	Frame	Ones
12	Secondo frame	Twos
6	Quarto frame	Fours

Ci sono molti altri trucchi di animazione, come lo spostamento degli sfondi in parallasse per avere un senso di movimento, o far brillare una luce dietro ad una immagine con ritagli per creare un'illusione di scintillio.

Semplice Pi Camera Trap con Raspberry

Questo progetto fa un po' di riciclaggio, usando un contenitore per alimenti in plastica per avere chiusura ermetica e resistere alle intemperie, quindi forse appartiene anche alla sezione precedente di questo speciale. È piuttosto semplice costruirla tecnicamente, ma fa un ottimo uso del codice per funzionare correttamente.

Funziona attendendo un movimento prima di scattare foto o filmati, grazie a MotionEyeOS, software utilizzato principalmente per TVCC. I principi sono molto simili, però.

magpi.cc/cameratrap

animazione Stop-motion

Questo progetto dal sito Raspberry Pi Projects è molto semplice ma molto efficace, permettendo di fare scatti continui usando la pressione di un pulsante prima di combinarli tutti in un unico video, alla fine. C'è anche un progetto simile nella *Camera Guide*. Potrebbe essere una ottima idea creare il set e i personaggi utilizzando le sezioni di making artigianale e riciclo.

magpi.cc/stopmotion

Motion camera time-lapse

C'è già un progetto di fotografia time-lapse in *The Official Raspberry Pi Camera Guide*. Questa configurazione particolare fa un ulteriore passo in avanti, tuttavia, consentendo un movimento preciso come parte del time lapse.

Usa un mix di Arduino e Raspberry Pi, insieme ad un po' di codice, per controllare una telecamera e un motore per ottenere panoramiche e rotazioni fluide su un video in time-lapse. Dovrebbe essere facile da aggiornare con la HQ Camera per scatti più sorprendenti.

magpi.cc/motionlapse

PROGETTI Raspberry Pi Pico SEMPLICI

Non lasciarti intimidire dal tuo nuovo microcontrollore Raspberry Pi Pico. Ecco alcuni grandi progetti in fase di realizzazione con Pico. Di Lucy Hattersley

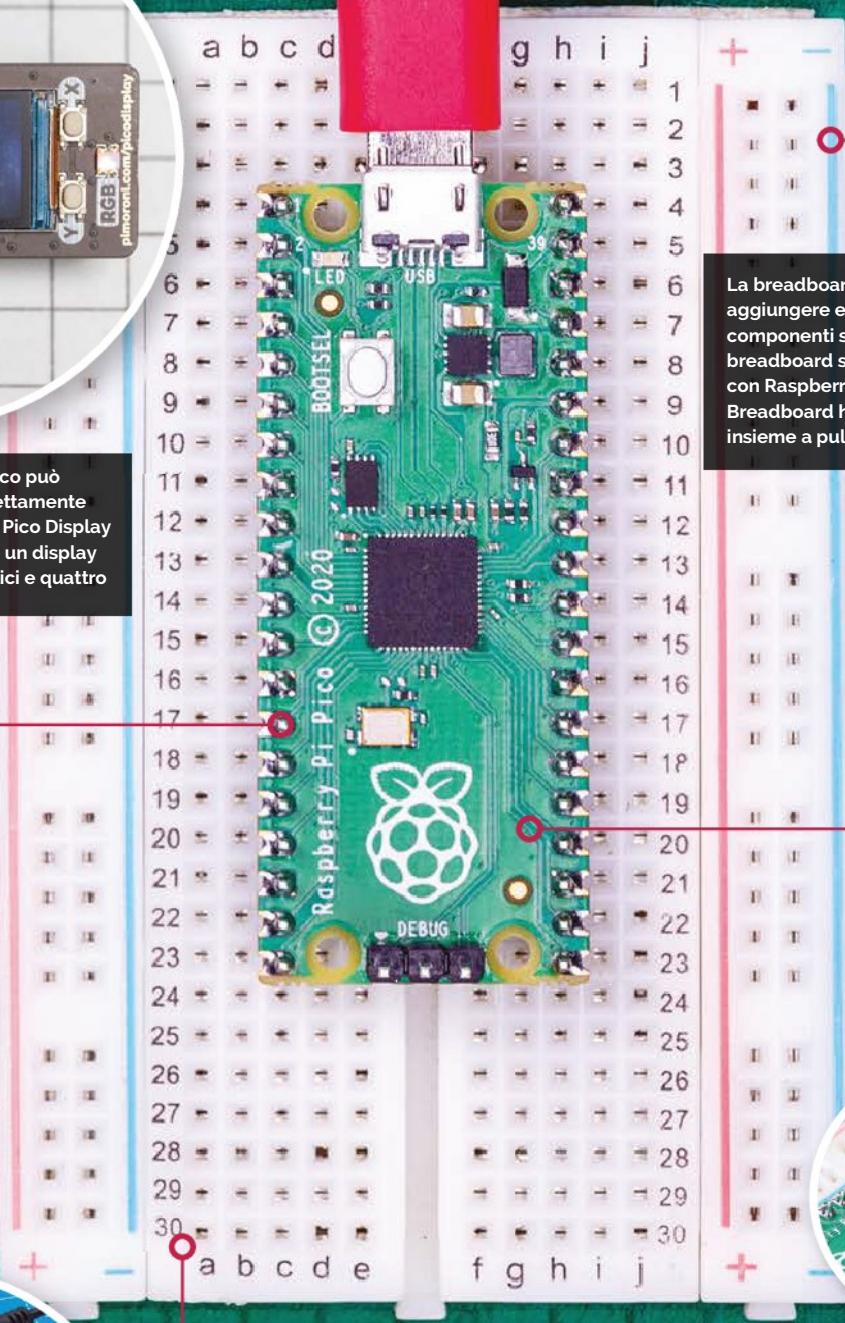
Raspberry Pi Pico è il nuovissimo microcontrollore di Raspberry Pi, che sta scuotendo il mondo della informatica. Abbiamo introdotto a fondo il Pico nell'ultimo numero (**bit.ly/MagPi102It**) e ci teniamo tanto quanto te a fare in modo che agguanti un Pico e inizia a fare cose. I microcontrollori sono una nuova tecnologia per Raspberry Pi e la rivista *MagPi* e la loro programmazione è leggermente diversa da quella per un Raspberry Pi con Raspberry Pi OS.

Fortunatamente, ci sono molti utenti che rendendo la vita con Pico molto più facile. Stanno creando una vasta gamma di prodotti Pico e add-on. Questi possono essere utilizzati per aggiungere schermi, luci, pulsanti e audio a Raspberry Pi Pico. Insieme agli esempi di codice e all'ottima documentazione, fanno esplodere il potenziale di Pico.

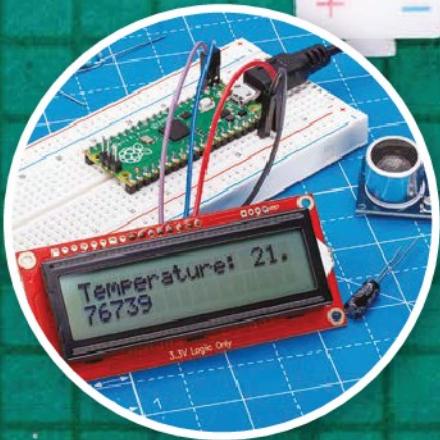
Ci sono pionieri che aprono la strada, con progetti ed esempi di come sfruttare al meglio Pico, con i suoi Pin GPIO e l'interessante tecnologia PIO (Programmable Input/Output).

In questo speciale, esamineremo alcune delle cose che gli utenti stanno facendo con Pico, e alcune semplici idee per progetti da realizzare.

> Ci sono molti utenti che rendono la vita con Pico molto più facile

SUGGERIMENTO

SALDARE I PIEDINI

Per provare i progetti su Raspberry Pi Pico e utilizzare la maggior parte dei kit menzionati in questo speciale, è necessario saldare dei pin sul tuo Raspberry Pi Pico. Vedi *The MagPi* N.102 (**bit.ly/MagPi102It**) per una guida sulla saldatura. In alternativa, Pimoroni o SB Components vendono entrambi dispositivi Raspberry Pi Pico con connettori pre-saldati.

La breadboard viene utilizzata per aggiungere e rimuovere rapidamente i componenti senza saldature. Alcune breadboard sono su misura per l'utilizzo con Raspberry Pi Pico. Il kit Pico Breadboard ha pin GPIO per il Pico saldato insieme a pulsanti, luci LED e un cicalino

La funzionalità di Pico può essere espansa direttamente con add-on come il Pico Display Pack, che aggiunge un display LCD IPS da 1,14 pollici e quattro pulsanti

I componenti elettronici, come lo SparkFun SerLCD display e i sensori nella foto possono essere testati nella breadboard, e poi cablati direttamente a Pico

I pin saldati ti consentono di attaccare Raspberry Pi Pico a una breadboard per la sperimentazione, e anche collegare nuovi add-on hardware progettati per Pico. Pimoroni e SB Components vendono schede Pico con connettori pre-saldati, se vuoi saltare il processo

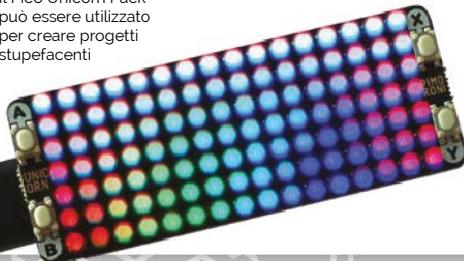
OFFRI VISUALE CON I TUOI PROGETTI

Dai vita al Pico con questi progetti visivi e sonori

I computer Raspberry Pi hanno uscite HDMI e prese mini jack da 3,5mm per il collegamento di cuffie o altoparlanti. Questo rende semplice ottenere feedback visivi e sonori dai tuoi programmi.

Raspberry Pi Pico è più essenziale. È possibile creare un'uscita DVI utilizzando i PIO su Pico, ma la maggior parte dei nuovi arrivati inizia facendo lampeggiare il led a bordo e invia il classico "Ciao Mondo" a un terminale su un altro computer (**magpi.cc/hellopico**). Oltre a questo, vorrai iniziare ad aggiungere kit elettronici per ottenere il massimo dal Pico.

Strisce pixel

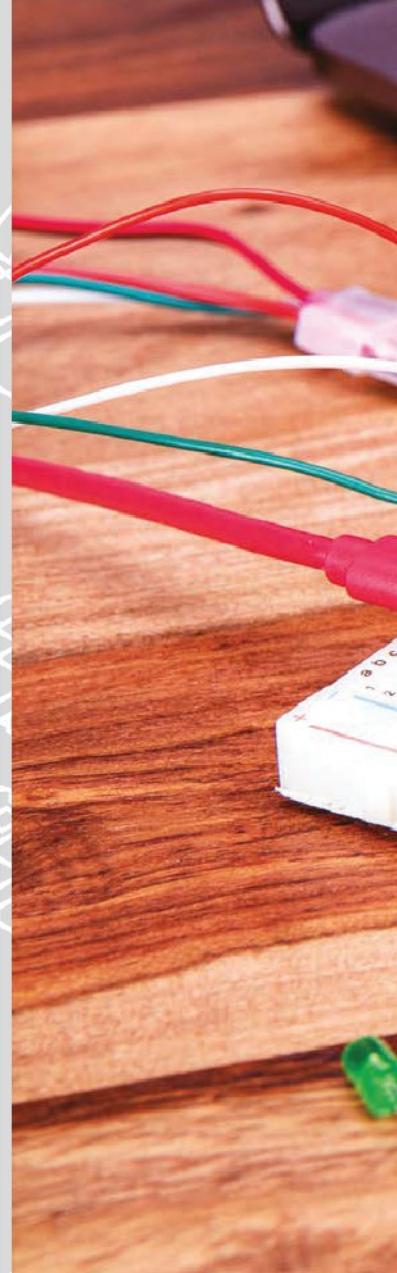
Se vuoi creare uno spettacolo di luci, i NeoPixel (noti anche come LED WS2812B) sono strisce economiche e facilmente reperibili (**magpi.cc/neopixels**).
Ogni striscia contiene una matrice di LED rossi, verdi e blu, progettati per essere controllati da un microcontrollore. Come ti aspetteresti, Pico è perfetto per controllare i display luminosi NeoPixel.

Ben Everard della rivista HackSpace ha scritto un articolo per Raspberry Pi sull'utilizzo di NeoPixel con Pico (**magpi.cc/neopixelpico**) e puoi dare un'occhiata al codice su GitHub (**magpi.cc/picolightsgit**).

Add-on visuali

Se vuoi qualcosa di personalizzato per il Pico, una serie di display luminosi, piccoli schermi e schede sonore, sono già in vendita. Pochi altri hanno supportato, fin'ora, la piattaforma Pico come Pimoroni (**magpi.cc/pimoronipico**).

Pimoroni ha realizzato due add.on a LED: l'Unicorn Pack è una matrice di LED a colori 16x7 (**magpi.cc/picounicorn**) e Scroll Pack (**magpi.cc/picoscroll**) è una matrice di LED 17×7 bianchi. I'Unicorn Pack può essere utilizzato per creare un display arcobaleno per la tua finestra (**magpi.cc/rainbow**), oltre a semplici animazioni. Lo Scroll Pack può essere utilizzato per creare messaggi di testo a scorrimento (da cui il nome) e è anche ottimo per visualizzare dati grafici e per creare una spia di stato.

In questo numero abbiamo il nostro primo progetto che utilizza un Pimoroni Pico Pack. A pagina 40 (non compreso nella traduzione), impara a usare un Unicorn Pack per costruire un Timer Pomodoro (uno strumento di produttività).

L'illuminazione d'atmosfera è un altro ottimo utilizzo per i display a luce LED; i LED bianchi di Scroll Pack o di Unicorn Pack possono essere utilizzati per illuminare le aree intorno alla casa, o per gettare una macchia di colore in un area specifica, come una vetrinetta.

Se vuoi qualcosa di più grande, il progetto PicoPythonHub75 (**magpi.cc/picohub75**) fa rimbalzare la parola "Pico" su e giù su un Pannello LED 32×32 RGB Hub75. I pannelli Hub75 sono un modo conveniente per aggiungere molte luci colorate a una build e l'output PIO di Pico consente animazioni super veloci.

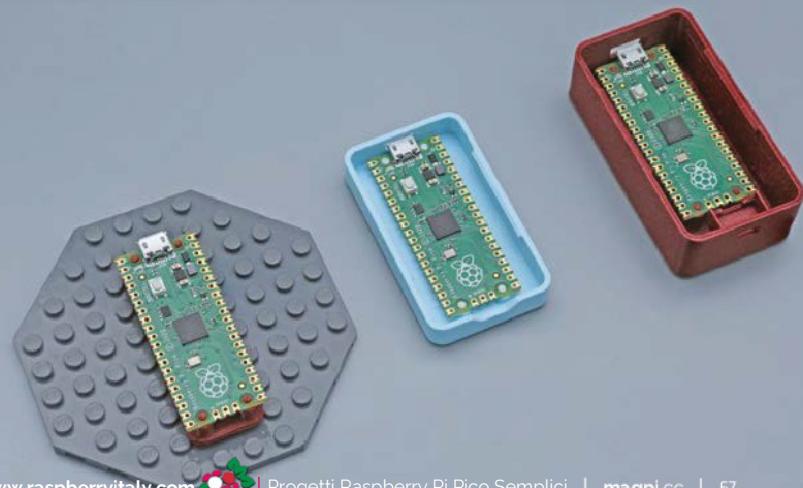
I display a LED sono presenti in una vasta gamma di progetti Pi, in parte perché un piccolo display a LED 8×8 RGB è stato incluso nel Sense HAT per Raspberry Pi. La pagina dei progetti di Sense HAT (**magpi.cc/sensehatprojects**) è piena di idee che potrebbe essere facilmente adattate a un Unicorn Pack o a un display a LED Hub75. Ogni cosa, da un timer con conto alla rovescia per una palla magica "8", al calendario dell'avvento e il diario meteo, potrebbe essere adattati da Raspberry Pi a Raspberry Pi Pico.

Il Pico Unicorn Pack può essere utilizzato per creare progetti stupefacenti

Aggiungere display

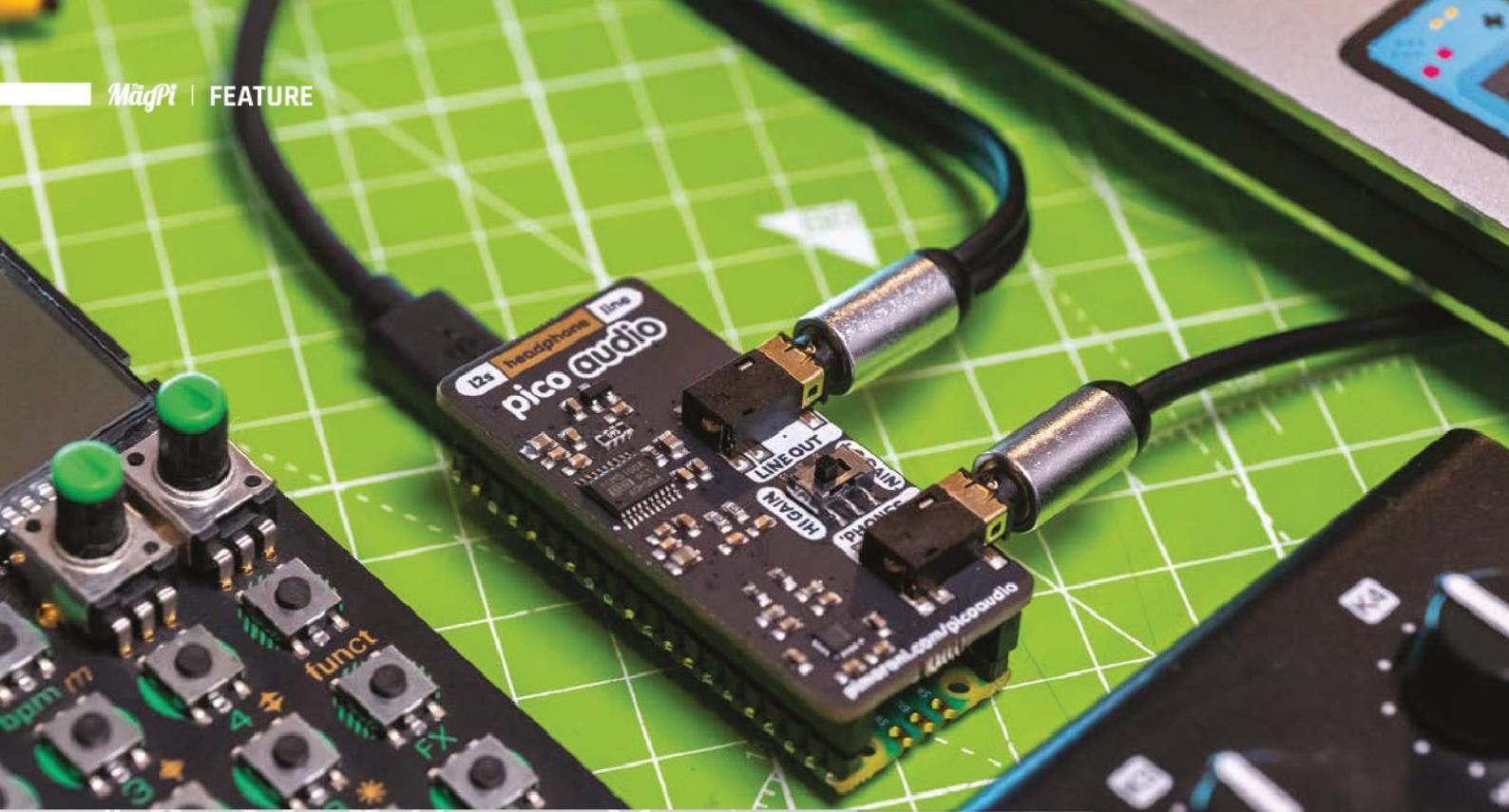
L'impressionante Pico Display Pack di Pimoroni (**magpi.cc/picodisplaypack**) aggiunge un piccolo display LCD IPS da 1,4 pollici alla scheda, con quattro pulsanti e un singolo LED RGB. Può essere utilizzato per visualizzare una piccola foto o brevi animazioni e fornire dei feedback sui dati. Puoi creare un termometro digitale utilizzando il sensore di temperatura interno di Pico (vedi il capitolo 8 del libro *Get Started with MicroPython on Raspberry Pi Pico*, **magpi.cc/picobook**).

Una volta che hai uno schermo su Pico, puoi collegare i sensori e visualizzare l'uscita, aprendo

Crea un'illuminazione D'atmosfera controllata da codice con Raspberry Pi Pico e NeoPixel

> Se vuoi creare uno spettacolo di luci, i NeoPixel sono economici e ampiamente disponibili

le possibilità a una gamma di progetti IoT (Internet of Things) e domestici. Ne esaminiamo alcuni nei i progetti elettronici più tardi, ma se collegherai dei sensori a Raspberry Pi Pico, uno schermo per mostrare un feedback è un ottimo componente aggiuntivo.

Molti dei tutorial di gioco più avanti in questo speciale, fanno buon uso del Pimoroni Display Pack. Se preferisci collegare direttamente un piccolo display, è relativamente facile collegare a Raspberry Pi Pico un SerLCD SparkFun (**magpi.cc/serlcd**). Puoi collegarlo all' I2C (Inter-Integrated Circuit) o all'interfaccia SPI (Serial Peripheral Interface) su Pico e ci sono metodi di controllo implementati in MicroPython. Questi piccoli display possono fornire feedback da altri dispositivi elettronici. Un buon progetto iniziale è visualizzare la temperatura del termometro integrato nel Pico. Dai un'occhiata al capitolo 10 di *Get Started with MicroPython on Raspberry Pi Pico* (**magpi.cc/picobook**).

CASE STAMPATO IN 3D PER RASPBERRY PI

Adafruit ha creato un supporto compatibile con Lego per Raspberry Pi Pico che aggiunge gli incastri alla nostra scheda microcontrollore preferita. Si adatta ai mattoncini standard o piastre di base, che ti consentono di applicare Pico su un progetto Lego (**magpi.cc/picolegocase**).

Il supporto è dotato di distanziatori incorporati in modo da poterci montare a scatto il Pico. Portando il progetto oltre, i Ruiz Bros hanno ha realizzato una custodia Pico con luci, pulsanti, un interruttore a slitta e una piccola Batteria LiPo. Puoi scaricare i file STL e vedere le istruzioni su Sito web di Adafruit (**magpi.cc/pico3dprintedcase**).

PROGETTI AUDIO

Aggiungi qualche suono ai tuoi progetti

Pico Audio Pack

Il Pico RGB Keypad può essere combinato con l'Audio Pack per creare dispositivi che rispondono ai tocchi delle dita. Perfetto per DJ e video streamer

Se parliamo di uscite audio, Raspberry Pi Pico è silenzioso. Non c'è nessun altoparlante integrato o nessuna presa mini jack da 3,5 mm. Tuttavia, non tutto è perduto. Ben Everard, dalla rivista HackSpace, la pubblicazione sorella di *The MagPi*, ha prodotto un eccellente tutorial PIO Buzz (**magpi.cc/piobuzz**) che spiega come collegare un altoparlante o delle cuffie ai PIO (Programmable Input/Output). I PIO del Pico consentono agli sviluppatori di definire nuove caratteristiche hardware nel software, espandendone le capacità al di là di qualsiasi dispositivo a funzioni fisse.

Ben dice che è "un semplice programma PIO che emette un tono basato su un numero inserito in esso. Non ho ancora fatto i conti per capire quale numero suoni quale nota, ma sembra offrire una bella gamma di toni udibili. Un po' ronzante alle frequenze più basse, ma suona meglio con le più alte".

Potrebbe non essere possibile riprodurre brani musicali, ma PIO Buzz è un buon modo per aggiungere del feedback audio ai tuoi progetti. I toni possono essere utilizzati per creare avvisi, notifiche e fornire feedback sull'interfaccia utente nei progetti.

Riproduzione audio

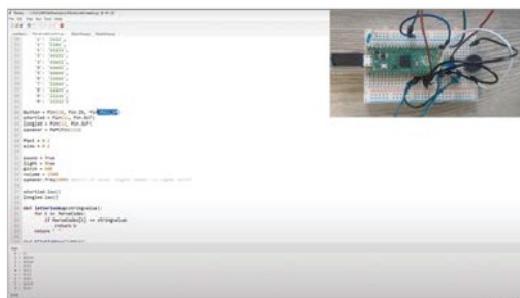
Se sei interessato alla riproduzione di file audio, il Pico Audio Pack (**magpi.cc/picoaudiopack**) sarà di tuo interesse. Aggiunge un DAC PCM 5100A (convertitore da digitale ad analogico) per emettere audio stereo fino a 32 bit, 384 kHz tramite un connettore line-out da 3,5 mm. Può anche pompare audio stereo amplificato da un jack per cuffie da 3,5 mm.

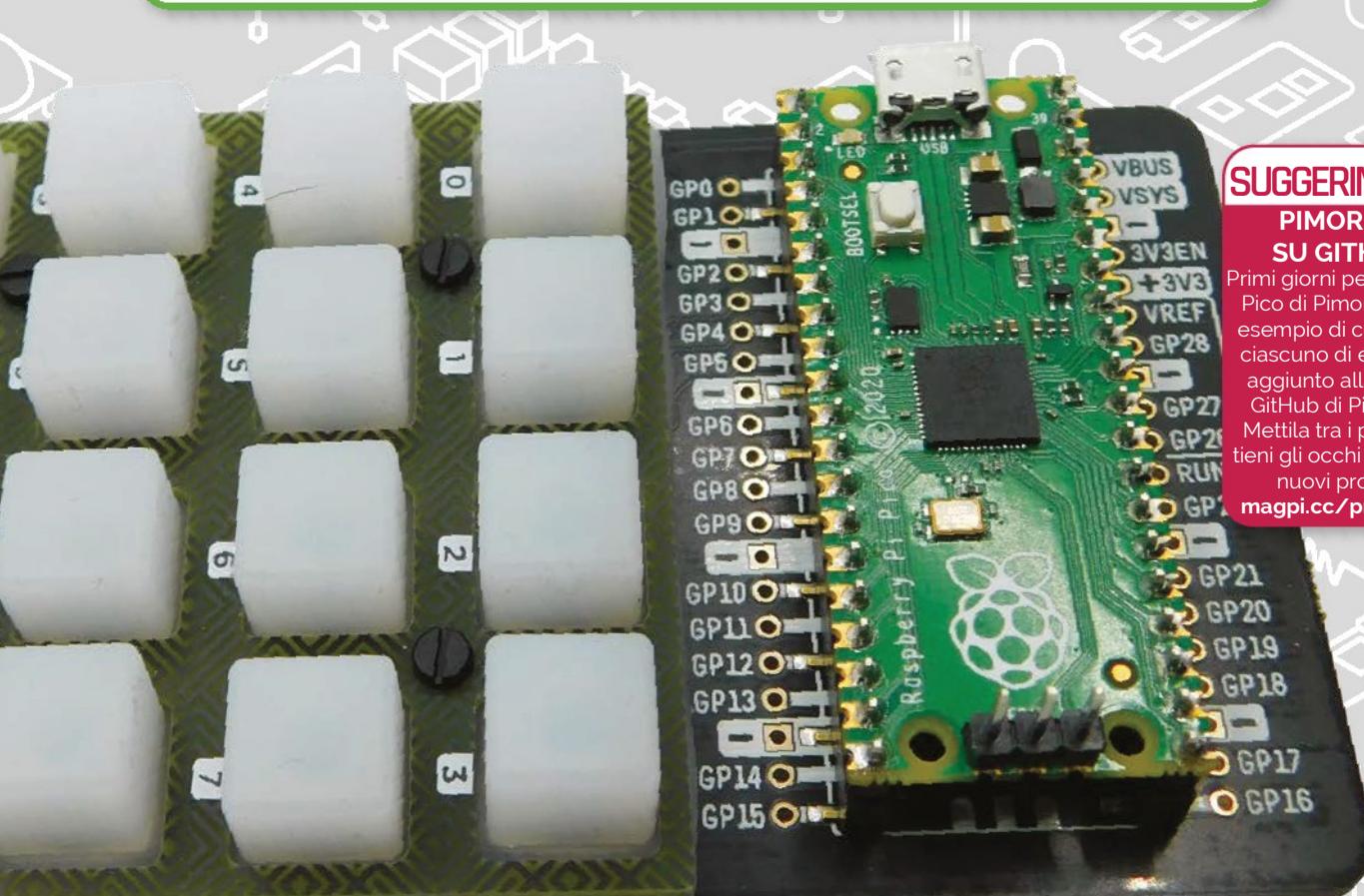
Con il Pico Audio Pack, puoi trasformare Raspberry Pi in un sintetizzatore lo-fi e generare rumori interessanti nel codice. Se stai creando progetti con il sintetizzatore, potresti anche voler guardare Pico RGB Keypad Base (**magpi.cc/picorgbkeyboard**). Questa è una griglia 4×4 righe di tasti illuminati che possono essere utilizzati per creare un dispositivo tipo tastiera USB. Sono popolari tra video streamer e DJ. Usalo per "Creare una pista da discoteca con le tue dita" dice Pimoroni, "o un gioco in stile Simon Says con cui scherzare con i tuoi amici".

> "Crea una pista da discoteca con le tue dita"

Una cosa da notare con il Pico Audio Pack è che funziona con l'SDK C/C++ di Pico (con il Supporto a MicroPython in arrivo). Quindi al momento adesso dovrai investire un po' più di tempo in un codice di base più impegnativo. Tuttavia, la possibilità di creare un lettore di musica digitale basato su Pico, un semplice gioco Simon, un sistema DJ o DAC per il tuo sistema audio domestico, vale l'investimento.

TRE FACILI PROGETTI PICO SU YOUTUBE

Print 'N Play è un canale YouTube dedicato a costruzioni e making. Illustra l'impostazione di Raspberry Pi Pico e passa a una serie di facili progetti di elettronica. Il codice per ciascun progetto è disponibile su GitHub.
magpi.cc/3easyprojects

▲ **Leggere la temperatura dal Pico**
magpi.cc/readtempgit

▲ **Suonare musica su Pico**
magpi.cc/playingmusicgit

▲ **Rirpodurre e registrare codice Morse**
magpi.cc/morsecodegit

SUGGERIMENTO
PIMORONI SU GITHUB

Primi giorni per i prodotti Pico di Pimoroni, e un esempio di codice per ciascuno di essi viene aggiunto alla pagina GitHub di Pimorioni. Mettila tra i preferiti e tieni gli occhi aperti per nuovi progetti.
magpi.cc/pmpicogit

PROGETTI PICO GAMES

Diventa creativo con i giochi Raspberry Pi Pico

I videogiochi sono un aspetto fondamentale per qualsiasi nuovo computer. A prima vista, potresti pensare che Pico non si presta ai giochi, ma ti sbaglieresti.

I PIO consentono di sollevare il controller principale dalla maggior parte dei compiti pesanti come il display, liberando il processore dual-core Arm Cortex M0+ per gestire il gioco. Il risultato è un sistema sorprendentemente versatile per i giochi.

Graham Sanderson ha dimostrato che è possibile emulare un BBC Micro e uno ZX Spectrum utilizzando i PIO per le uscite audio e la grafica VGA. Non ci sono istruzioni ancora, ma puoi controllare la build su YouTube (**magpi.cc/picobbcmicro**).

Se stai cercando una build con codice pronto da far girare, dai quindi un'occhiata a Tetris per Pico, costruito utilizzando il Pico Display Pack (**magpi.cc/picodisplaypack**). Usa i pulsanti di quest'ultimo per ruotare e controllare i pezzi, e il piccolo schermo mostra il gioco. Il codice per Pico Tetris può essere trovato su PasteBin (**magpi.cc/picotetris**).

Il piccolo schermo del Pico Display Pack o i LED del Pico Unicorn Pack si prestano ai giochi in stile Tamagotchi, e puoi facilmente crearne uno per Raspberry Pi Pico utilizzando delle animazioni. C'è il codice Pixel Pet per Raspberry Pi Sense HAT sulla pagina dei progetti (**magpi.cc/pixelpet**) che potrebbe essere facilmente convertito per Raspberry Pi Pico e un display Pico Pack o Unicorn Pack.

> Se stai cercando una build con codice pronto a girare, dai uno sguardo a Tetris for Pico

Il computer BBC Micro emulato su Raspberry Pi Pico, con gestione PIO dell'uscita display

Se stai cercando un sistema da gioco più completo, PicoSystem di Pimoroni (**magpi.cc/picosystem**) è in fase di sviluppo. PicoSystem promette di essere una minuscola console di gioco portatile tutta-in-uno costruita intorno al microcontrollore RP2040, il cuore di Pico. Dispone di un piccolo schermo LCD IPS, joypad, pulsanti, e batteria LiPo.

Raspberry Pi Pico si sta dimostrando abbastanza potente per i giochi retrò, molto di più di quanto potresti immaginare.

PicoSystem è una console di gioco tutta-in-uno basata sul potente microcontrollore RP2040, il cuore di Raspberry Pi Pico

ELETTRONICA FACILE CON PICO

Crea i tuoi progetti con componenti e codice

Come tutte le schede Raspberry Pi, Pico è eccellente per l'apprendimento dell'elettronica e per realizzare piccoli circuiti che fanno ogni genere di cose. È possibile saldare fili e componenti direttamente ai socket GPIO, ma l'aggiunta di pin di intestazione consente di inserire Pico nei fori di una breadboard (**magpi.cc/breadboard**) e di collegare e scollegare i componenti senza doverli saldare.

> "Non sarai mai a corto di progetti da provare"

Con i pin sul Pico, puoi seguire la guida Guida "Getting started with Raspberry Pi Pico" (**magpi.cc/gettingstartedpico**). E non perderti una copia di *Get Started with MicroPython on Raspberry Pi Pico* (**magpi.cc/picobook**). Troverai un tutorial sull'uso di Pico con LED e pulsanti anche qui, nelle pagine seguenti.

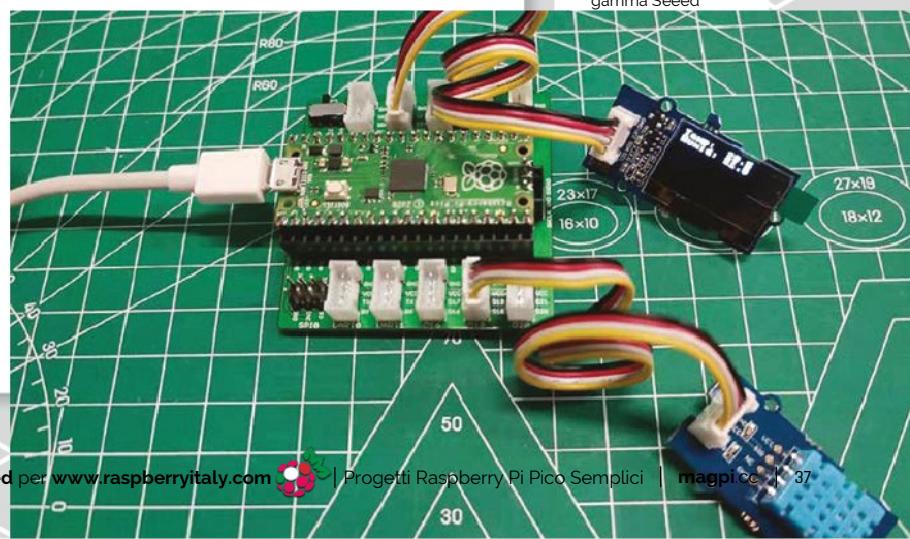
Elettronica facile

Sono stati sviluppati numerosi dispositivi per aiutare i test elettronici e lo sviluppo con Raspberry Pi Pico. SB Components ha prodotto un Pico Breadboard Kit (**magpi.cc/picobread**) che dovrebbe rendere più facile la prototipazione. Anche come una breadboard di mezza dimensione, racchiude un cicalino, quattro LED e quattro pulsanti. Ha anche pin dedicati 5 V, 3V3 e GND.

Un altro dispositivo che mira a rendere più accessibile l'elettronica è il Pico Explorer Base (**magpi.cc/picoexplorer**). Questo include anche una mini breadboard, ma anche altre schede con uno schermo LCD IPS 240×240 e quattro pulsanti (per creare menu interattivi per i tuoi progetti). Ha anche due slot Breakout Garden che puoi utilizzare per inserire rapidamente i dispositivi di Pimoroni breakout range (**magpi.cc/pimoronibreakouts**). Questi includono tutto, dai sensori per la qualità dell'aria ai pulsanti con feedback tattile.

Se vuoi evitare un groviglio di fili e sperimentare con una serie di componenti, dai un occhio a Grove Shield di Seeed per Raspberry Pi Pico (**magpi.cc/groveshield**). Questo ti permette di collegare plug and play oltre 300 moduli Grove. Seeed ha fornito una guida dettagliata all'uso del Grove Shield con Raspberry Pi Pico (**magpi.cc/groveshieldtutorial**) che mostra come collegare un cicalino, un sensore di angolo rotante, un display OLED e un sensore di temperatura con Raspberry Pi Pico.

HAT ovunque

Un altro dispositivo è il Raspberry Pi Pico HAT Expansion (**magpi.cc/picohatexpansion**). È una scheda di input/output che trasforma i pin Raspberry Pi Pico nel connettore a 40 pin che si trova sui computer Raspberry Pi. Ciò consente di collegare uno qualsiasi delle miriadi di HAT progettati per Raspberry Pi, al Pico (anche se vale la pena sottolineare che il software e le API progettati per Raspberry Pi potrebbero necessitare di un lavoro di traduzione per funzionare in MicroPython o C/C++). Se hai un Progetto su Raspberry Pi maturo per la conversione per Pico, questo hardware potrebbe essere la cosa giusta.

Pico Omnibus Dual Expander di Pimoroni (**magpi.cc/picoomnibus**) ti permette di raddoppiare le tue espansioni, aggiungendone due contemporaneamente. Oppure puoi utilizzare i pin GPIO aggiuntivi per il collegamento di ponticelli o circuiti insieme a un Pico Pack. Se ciò non bastasse, il Pico Decker Quad Expander (**magpi.cc/picodecker**) offre supporto per un massimo di quattro dispositivi Pico Pack. Sulla scheda c'è una "zona di atterraggio" con connettori femmina serigrafati per accogliere il Pico e quattro ulteriori aree di atterraggio con connettori specchiati per Inserirvi gli add-on.

Una cosa è certa: con Pico non sarai mai a corto di progetti da provare.

RICONOSCERE LE PERSONE CON LA IA

Ecco un altro progetto che potrebbe essere adatto per i maker più avanzati. Kyle di Arducam ha fatto funzionare TensorFlow Lite Micro su Raspberry Pi Pico e ha creato un tutorial sull'utilizzo del machine learning per il rilevamento delle persone. Collega una videocamera al tuo Raspberry Pi Pico e sarai in grado di utilizzare il riconoscimento delle immagini per rilevare le persone.
magpi.cc/tfpico

Il Grove Shield per Raspberry Pi Pico ti consente di collegare oltre 300 moduli della gamma Seeed

Physical computing con Raspberry Pi Pico

Inizia a collegare i componenti elettronici di base al tuo Raspberry Pi Pico e a scrivere programmi per controllarli e rilevarli

SUGGERIMENTO

IMPORTAZIONI SELETTIVE

Sia in MicroPython che in Python, è possibile importare solo una parte di una libreria, piuttosto che completa. In questo modo puoi creare un programma che usa meno memoria, e che ti permette di mescolare e abbinare funzioni da librerie diverse. I programmi in questa guida importano l'intera libreria; altrove puoi vedere programmi che hanno righe tipo `from machine import Pin`; Questo dice a MicroPython di importare solo la funzione "Pin" dalla libreria "machine", piuttosto che l'intera libreria.

Raspberry Pi Pico, con il suo microcontrollore RP2040, è progettato tenendo ben in mente il physical computing. I suoi numerosi pin di input/output generici (GPIO) gli consentono di parlare con una varietà di componenti, permettendoti di realizzare progetti, dalla illuminazione a LED alla registrazione dei dati sul mondo che ti circonda.

Il Physical computing non è più difficile da imparare rispetto all'informatica tradizionale: se sai programmare un po' in Python, sarai in grado di crearne i tuoi circuiti e programmarli per eseguire i tuoi ordini.

Il tuo primo programma di physical computing: Ciao, LED!

Proprio come scrivere "Ciao Mondo" sullo schermo è un fantastico primo passo per imparare un linguaggio di programmazione, far accendere un LED è la tradizionale introduzione all'apprendimento del physical computing. Puoi anche iniziare senza componenti aggiuntivi: il tuo Raspberry Pi Pico ha un piccolo LED, a montaggio superficiale (SMD), nella parte superiore.

Inizia trovando il LED: è il piccolo componente rettangolare a sinistra della porta micro-USB nella parte superiore della scheda (Figura 1), contrassegnato con l'indicazione della scritta "LED". Questo piccolo LED funziona proprio come qualsiasi altro: quando è alimentato, si illuminerà; quando è spento, rimane buio.

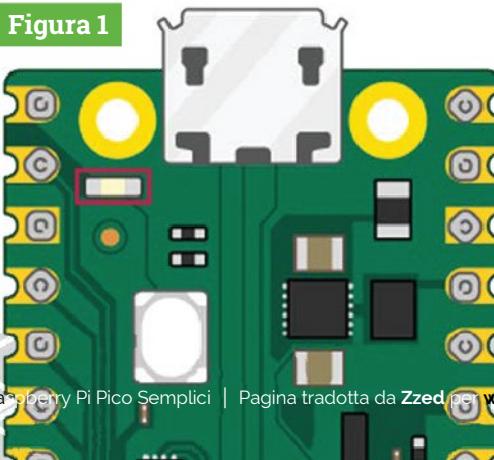



Figura 1 Il LED a bordo si trova a sinistra del connettore micro-USB

Il LED a bordo è collegato a uno dei pin di input / output per uso generico dell'RP2040, il GP25. Questo è uno dei pin GPIO "mancanti": presente sul RP2040 ma non portato ad un pin fisico sulla scheda del tuo Pico. Non puoi quindi collegare qualsiasi hardware esterno a questo pin, a parte il LED di bordo, ma può essere trattato come qualsiasi altro pin GPIO all'interno dei tuoi programmi - ed è un ottimo modo per aggiungere un output ai tuoi programmi, senza bisogno di componenti aggiuntivi.

Carica Thonny e, con Pico collegato al tuo Raspberry Pi, fai clic su "Python" in basso a destra, quindi seleziona "MicroPython (Raspberry Pi Pico)" come interprete. Fai clic nell'area dello script e comincia il programma con la seguente riga:

```
import machine
```

Questa breve riga di codice è la chiave per lavorare con MicroPython sul tuo Pico: carica, o *importa*, una raccolta di codice MicroPython noto come libreria - in questo caso, la libreria "machine". Questa libreria contiene tutte le istruzioni che MicroPython Necessita per comunicare con il Pico e altri dispositivi compatibili con MicroPython, estendendo il linguaggio per il physical computing. Senza questa riga non sarai in grado di controllare nessuno dei pin GPIO del Pico - e non sarai in grado di far accendere il LED a bordo.

La libreria machine espone ciò che è noto come *application programming interface* (API). Il nome sembra complicato, ma descrive esattamente cosa fa: fornisce un modo per il tuo programma, o *applicazione*, per comunicare con Pico tramite una interfaccia. La riga successiva del programma fornisce un esempio delle API della libreria machine:

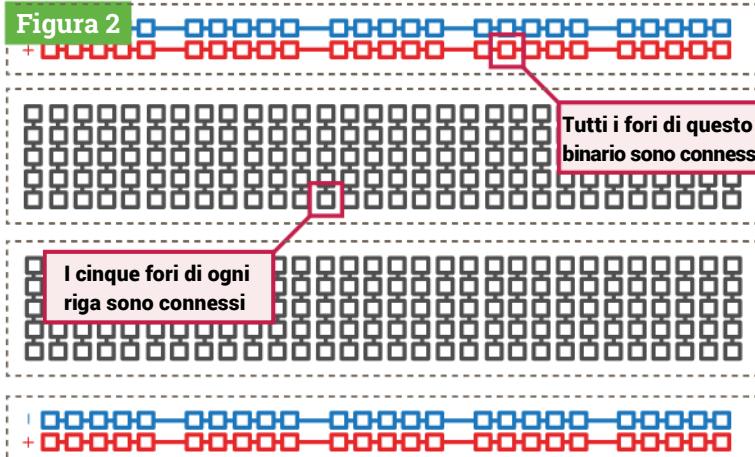
```
led_onboard = machine.Pin(25, machine.Pin.OUT)
```

Questa riga definisce un oggetto chiamato `led_onboard`,che offre un nome descrittivo che puoi utilizzare per fare riferimento al LED di bordo più avanti nel programma. È tecnicamente possibile usare qualsiasi nome – come susanna, gianni o sandwich_pesce, ma è meglio attenersi a nomi che descrivono lo scopo della variabile, per rendere il programma più facile da leggere e capire.

La seconda parte della riga, richiama la funzione Pin nella libreria machine. Questa funzione, come il suo nome suggerisce, è progettata per gestire i pin del GPIO di Pico. Al momento, nessuno dei pin GPIO - compreso GP25, il pin collegato al LED a bordo- sa cosa deve fare. Il primo argomento, `25`, è il numero del pin che stai impostando; il secondo, `machine.Pin.OUT`, dice a Pico che il pin deve essere usato come uscita, invece che come ingresso.

Quella riga, da sola, è sufficiente per impostare il pin a livello alto, ma così non accenderà il LED. Per farlo, devi dire al Pico di attivare effettivamente il pin. Digita il codice seguente, nella riga successiva:

```
led_onboard.value(1)
```

> Il LED a bordo è collegato a un pin GPIO del RP2040

Potrebbe non sembrare, ma anche questa riga utilizza l'API della libreria machine. La tua riga precedente ha creato l'oggetto `led_onboard` come uscita sul pin GP25; Questa riga prende l'oggetto e ne imposta il valore su 1 per "on" - potrebbe anche impostare il valore su 0, per "off".

Fai clic sul pulsante Esegui e salva il programma sul tuo Pico come **Blink.py**. Vedrai accendersi il LED. Congratulazioni, hai scritto il tuo primo programma di physical computing!

Noterai, tuttavia, che il LED rimane acceso: questo perché il tuo programma dice a Pico di accenderlo, ma non gli dice mai di spegnerlo. Puoi aggiungere un'altra riga in fondo al tuo programma:

```
led_onboard.value(0)
```

Esegui il programma, questa volta, però, il LED non sembra accendersi. Questo perché Pico funziona molto, molto rapidamente, molto più rapidamente di quanto puoi vedere ad occhio

Figura 2 Le connessioni interne di una breadboard

nudo. Il LED si accende, ma per così poco tempo che sembra rimanere spento. Per per risolvere questo problema, è necessario rallentare il programma, introducendo un *ritardo*.

Torna all'inizio del programma: fai clic alla fine della prima riga e premi **INVIO** per inserire una nuova seconda riga. Su questa riga, digita:

```
import utime
```

Come `import machine`, questa riga importa una nuova libreria in MicroPython: la libreria "utime". Questa libreria gestisce tutto ciò che ha a che fare con il tempo, dalla misurazione di questo, all'inserire ritardi nei tuoi programmi.

Vai in fondo al programma e fai clic sulla fine della riga `led_onboard.value (1)`, quindi premi **INVIO** per inserire una nuova riga. Digita:

```
utime.sleep(5)
```

Questo richiama la funzione sleep dalla libreria utime, che mette in pausa il tuo programma per il numero di secondi che hai digitato, in questo caso, cinque secondi.

Fai di nuovo clic sul pulsante Esegui. Questa volta vedrai il LED di bordo del tuo Pico accendersi, restare acceso per cinque secondi - prova a contarli – e spegnersi di nuovo.

Infine, è il momento di far lampeggiare il LED. Per farlo, dovrai creare un ciclo. Riscrivi il tuo programma in modo che corrisponda a quello qui sotto:

```
import machine
import utime

led_onboard = machine.Pin(25, machine.Pin.OUT)
```

SUGGERIMENTO

NUMERAZIONE PIN

Quando si parla di pin GPIO sul tuo Pico, sono di solito indicati utilizzando il loro nome completo: GP25 per il pin collegato al LED a bordo, per esempio. In MicroPython, però, le lettere G e P si sono perse - così accertati di scrivere "25" anziché "GP25" nel tuo programma, o non funzionerà!

Figura 3 Il Pico è progettato per inserirsi in modo sicuro su una breadboard

```
while True:
    led_onboard.value(1)
    utime.sleep(5)
    led_onboard.value(0)
    utime.sleep(5)
```

Ricorda che le righe all'interno del ciclo devono essere rientrate di quattro spazi, così MicroPython sa che formano un ciclo. Fai di nuovo clic sull'icona Esegui e vedrai il LED accendersi per cinque secondi, spegnersi per cinque secondi e poi riaccendersi, ripetendo un ciclo infinito. Il LED continuerà a lampeggiare finché non fai clic sull'icona Ferma per cancellare il programma e resettare il Pico.

C'è anche un altro sistema per ottenere lo stesso scopo: utilizzando *toggle*, anziché impostare l'uscita del LED esplicitamente a 0 o 1. Elimina le ultime quattro righe del programma e sostituiscile in modo che appaia così:

```
import machine
import utime

led_onboard = machine.Pin(25, machine.Pin.OUT)

while True:
    led_onboard.toggle()
    utime.sleep(5)
```

Esegui di nuovo il programma. Vedrai lo stesso risultato di prima: il LED a bordo si accenderà per cinque secondi, poi si spegnerà per cinque secondi, quindi si accenderà di nuovo, in un ciclo infinito. Questa volta, però, il programma è più breve di due righe: l'hai *ottimizzato*. Disponibile su tutti i pin di uscita digitale, `toggle()` semplicemente commuta tra on e off: se il pin è attualmente on, `toggle()` lo spegne; se è off, `toggle()` lo accende.

SUGGERIMENTO

UTIME VS TIME

Se hai già programmato in Python, sarai abituato a usare la libreria "time". La libreria utime è una versione progettata per microcontrollori come il Pico - la "u" che sta per "µ", la lettera greca "mu", che viene utilizzata come abbreviazione per "micro". Se ti dimentichi e usi `import time`, non preoccuparti: MicroPython automaticamente userà, invece, la libreria utime.

Usare una breadboard

I seguenti progetti in questo capitolo saranno molto più facili da completare se utilizzi una breadboard per assemblare i componenti e realizzare le connessioni elettriche.

Una breadboard (**Figura 2**) è ricoperta da fori - distanziati, per collegare i componenti, a 2,54 mm di distanza. Sotto questi fori ci sono strisce di metallo che agiscono come i cavi jumper che hai utilizzato fino a ora. Questi corrono in file lungo tutta la basetta, con la maggior parte delle schede che hanno uno spazio nel mezzo per dividerle in due metà.

Molte breadboard hanno lettere sulle righe in alto e numeri lungo i lati. Questi ti permettono di trovare un foro particolare: A1 è l'angolo in alto a sinistra, B1 è il foro immediatamente a destra, mentre B2 è una riga più in giù. A1 è collegato a B1 dalla striscia di metallo nascosta, ma nessun foro contrassegnato con 1 è collegato a nessun foro contrassegnato con un 2 a meno che tu stesso non aggiungi un ponticello.

> "Aggiungere componenti elettronici a una breadboard è facile"

Le breadboard più grandi hanno anche strisce di fori lungo i lati, tipicamente contrassegnati con strisce rosse e nere o rosse e blu. Questi sono i binari di alimentazione e sono progettati per rendere più facile il cablaggio: puoi connettere un singolo filo da uno dei pin di massa del tuo Pico a uno dei binari di alimentazione, tipicamente contrassegnato con una striscia blu o nera e un simbolo meno - per fornire una massa comune per molti componenti sulla breadboard, e puoi fare lo stesso se il tuo circuito necessita di alimentazione a 3,3V o 5V. Nota: tutti i fori uniti alle strisce sono collegati tra loro; uno spazio vuoto indica una interruzione.

Aggiungere componenti elettronici a una breadboard è facile: basta allineare i loro terminali (i piedini metallici sporgenti) con i fori e spingere delicatamente fino a quando il componente entra al suo posto. Per i collegamenti è necessario oltre a quelli che la breadboard fa per te, utilizzare cavi jumper maschio-maschio (M2M); per collegamenti della breadboard verso i dispositivi esterni, come un Raspberry Pi Pico, utilizzano cavi ponticello maschio-femmina (M2F).

Non tentare mai di stipare più di un terminale di componente o ponticello in un singolo foro sulla breadboard. Ricorda: i fori sono collegati in file, a parte la divisione al centro, quindi al terminale del componente in A1 è collegato elettricamente a tutto ciò che aggiungi a B1, C1, D1 ed E1.

Inserisci il Pico sulla breadboard in modo che sia a cavallo del separatore centrale e con la porta micro USB verso la parte superiore della scheda (vedi la **Figura 3**). Il pin in alto a sinistra, Pin 0, dovrebbe essere nella riga della breadboard contrassegnata con A1, se la breadboard è numerata. Prima di spingere il Pico giù, assicurati che i pin siano tutti posizionati correttamente: se pieghi un pin può essere difficile raddrizzarlo nuovamente, senza che si rompa.

Spingere delicatamente il Pico verso il basso finché le parti in plastica dei pin toccano la breadboard. Ciò significa che le parti metalliche dei pin sono completamente inserite e hanno un buon contatto elettrico con la breadboard.

Prossimo passo: un LED esterno

Finora hai lavorato solo con il Pico - eseguendo programmi MicroPython sul microcontrollore RP2040 e commutando il LED a bordo tra acceso e spento. Di solito i microcontrollori vengono utilizzati, però, con componenti esterni, e il tuo Pico non fa eccezione.

Per questo progetto, avrai bisogno di una breadboard, ponticelli maschio-maschio (M2M), un LED e una resistenza da 330Ω - o il più vicino a 330Ω che hai a disposizione. Se non hai una breadboard, puoi utilizzare cavi jumper femmina-femmina (F2F), ma il circuito sarà fragile e facile da rompere.

Tieni il LED tra le dita: vedrai uno dei suoi terminali più lungo dell'altro. Il terminale più lungo è noto come *anodo* e rappresenta il positivo del circuito; il terminale più corto è il *catodo*, e rappresenta il negativo. L'anodo deve essere collegato a uno dei pin GPIO del Pico tramite la resistenza; il catodo deve essere collegato ad un pin di massa.

Inizia collegando la resistenza: prendi un'estremità (non importa quale) e inseriscila nella breadboard nella stessa riga del pin GP15 del Pico in basso a sinistra, se utilizzi una breadboard numerata con il Pico inserito in alto, questa dovrebbe essere la riga 20. Spingere l'altra estremità dentro una riga libera più in basso nella breadboard: noi stiamo utilizzando la riga 24.

Prendi il LED e spingi la gambetta più lunga - l'anodo - nella stessa riga dell'estremità della resistenza. Spingere la gambetta più corta – il catodo - nella stessa riga ma attraversando lo spazio centrale nella breadboard, quindi è allineato ma non collegato elettricamente al terminale più lungo, tranne che tramite il LED stesso. Infine, inserire un ponticello maschio-maschio (M2M) nella stessa riga del terminale più corto del LED, quindi collegalo direttamente a uno dei pin di massa del Pico (tramite un altro foro nella sua riga) o sul binario negativo della tua breadboard. Se lo colleghi al binario di alimentazione, termina il circuito collegando quello negativo a uno dei piedini di massa del Pico. Hai finito il circuito che dovrebbe assomigliare alla **Figura 4**.

Il controllo di un LED esterno in MicroPython non è diverso dal controllo del LED interno del Pico: cambia solo il numero del pin. Se hai chiuso Thonny, riaprilo, e carica il programma **Blink.py** dell'inizio del capitolo. Trova la riga:

```
led_onboard = machine.Pin(25, machine.Pin.OUT)
```

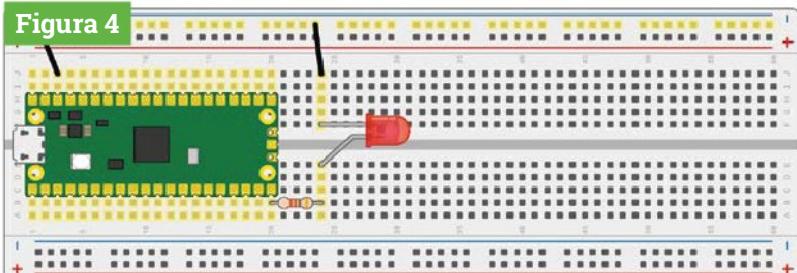
Modifica il numero del pin, cambiandolo da 25 - il pin collegato al LED interno del Pico - a 15, il pin a cui hai collegato il LED esterno. Modifica anche il nome che hai creato: non stai usando il LED a bordo, quindi chiamalo **`led_external.`** Dovrai anche cambiare il nome nel programma, finché non assomiglia a questo:

```
import machine
import utime

led_external = machine.Pin(15, machine.Pin.OUT)

while True:
    led_external.toggle()
    utime.sleep(5)
```

SUGGERIMENTO

LA RESISTENZA È VITALE

La resistenza è un componente vitale in questo circuito: protegge sia il Pico che il LED, limitando la quantità di corrente elettrica che attraversa il LED. Senza di essa, il LED potrebbe assorbire troppa corrente e bruciarsi - o bruciare il Pico. Quando viene utilizzata in questo modo, la resistenza è nota come resistenza limitatrice di corrente. Il valore esatto della resistenza necessaria dipende dal LED che stai utilizzando, ma 330Ω funziona per i LED più comuni. Più alto è il valore, più la luce del LED si attenua; minore è il valore, più luminoso è il LED. Non collegare mai un LED al Pico senza una resistenza limitatrice di corrente, a meno che tu non sappia che il LED ha una resistenza incorporata di valore appropriato.

Figura 4 Il circuito finito, con un LED e una resistenza

SUGGERIMENTO

CONVENZIONI NEI NOMI

Non è davvero necessario modificare il nome oggetto LED nel programma: funzionerebbe allo stesso modo se lo lasciassi a led_onboard, poiché è solo il numero del pin che conta veramente. Quando torni al programma In un secondo momento, però, sarebbe fonte di confusione avere un oggetto denominato led_onboard che accende un LED esterno – così prova a prendere l'abitudine di usare nomi che descrivono il loro scopo!

Ingressi: leggere un pulsante

Gli output tipo i LED sono una cosa, ma la parte "input/output " di "GPIO" comprende anche un utilizzo dei pin come input. Per questo progetto, avrai bisogno di una breadboard, cavi jumper maschio-maschio e un pulsante. Se non hai una breadboard, è possibile utilizzare cavi jumper femmina-femmina (F2F), ma il pulsante sarà molto più difficile da premere senza interruzioni accidentali del circuito.

Rimuovi qualsiasi altro componente dalla breadboard tranne il Pico, e inizia aggiungendo il pulsante. Se il pulsante ha solo due piedini, assicurati che siano su righe diverse della breadboard, da qualche parte sotto il Pico. Se ha quattro piedini, giralo in modo che i lati dove ci sono i terminali siano lungo le righe della breadboard e i lati piatti senza terminali siano in alto e in basso, prima di spingerlo in sede, in modo che si trovi a cavallo della divisione centrale della breadboard.

Collega il binario alimentazione positivo della breadboard al pin 3V3 del Pico e da lì a uno dei piedini del pulsante; quindi collega l'altro terminale al GP14 sul Pico: è quello appena sopra il pin che hai usato per il progetto LED, e dovrebbe essere nella riga 19 della breadboard.

> "La pressione del pulsante ha completato il circuito e cambiato il valore letto"

Se utilizzi un pulsante a quattro piedini, il circuito funziona solo se utilizzi la coppia corretta di piedini: questi sono collegati a coppie, quindi è necessario utilizzare i due sullo stesso lato o (come si vede in **Figura 5**) i piedini diagonalmente opposti.

Carica Thonny, se non l'hai già fatto, e avvia un nuovo programma con la solita riga:

```
import machine
```

Successivamente, dovrai utilizzare le API di machine per configurare un pin come input, invece che come output:

Figura 5 Cablaggio del pulsante a quattro piedini sul GP14

```
button = machine.Pin(14, machine.Pin.IN,
machine.Pin.PULL_DOWN)
```

Funziona allo stesso modo dei progetti con LED: viene creato un oggetto chiamato "button", che include il numero del pin – GP14, in questo caso – e lo configura come ingresso con la resistenza impostata come pull-down. Creare l'oggetto, però, non significa che qualsiasi cosa andrà bene da sola, proprio come creare gli oggetti LED in precedenza non accendeva i LED.

Per leggere effettivamente il pulsante, è necessario utilizzare di nuovo l'API machine, questa volta utilizzando la funzione `value` per leggere, piuttosto che impostare, il valore del pin. Digita la seguente riga:

```
print(button.value())
```

Fare clic su Esegui e salva il programma come **Button.py** – ricordando di fare attenzione che sia salvato sul Raspberry Pi Pico. Il programma mostrerà un singolo numero: il valore dell'ingresso GP14. Poiché l'input sta usando una resistenza in pull-down, questo valore sarà 0 – dicendoti che il pulsante non è premuto.

Tieni premuto il pulsante col dito e premi nuovamente l'icona Esegui. Questa volta vedrai il valore 1 mostrato nella Shell: la pressione del pulsante ha completato il circuito e cambiato il valore letto dal pin. Per leggere il pulsante in modo continuo, dovrai aggiungere un ciclo al tuo programma. Modifica il programma in modo che sia come di seguito:

```
import machine
import utime

button = machine.Pin(14, machine.Pin.IN,
machine.Pin.PULL_DOWN)

while True:
    if button.value() == 1:
        print("Hai premuto il pulsante!")
        utime.sleep(2)
```

Fai di nuovo clic sul pulsante Esegui. Questa volta, non accadrà niente fino a quando non premi il pulsante; quando lo fai, vedrai un messaggio visualizzato nella Shell. Il ritardo, è importante: ricorda, il Pico è molto più veloce di quanto tu possa leggere, e senza il ritardo anche una breve pressione del pulsante può visualizzare centinaia di messaggi sulla Shell!

Vedrai il messaggio ogni volta che premi il tasto. Se tieni premuto il pulsante più a lungo rispetto al ritardo di due secondi, visualizzerà il messaggio ogni due secondi fino a quando non rilasci il pulsante.

Ingressi e uscite: mettere tutto insieme

La maggior parte dei circuiti ha più di un componente, ecco perché il Pico ha così tanti pin GPIO. È tempo di mettere insieme tutto ciò che hai imparato per costruire un circuito più complesso: un dispositivo che accende e spegne un LED con un pulsante.

In effetti, questo circuito combina entrambi i circuiti precedenti, in uno. Potresti ricordarti di aver utilizzato il pin GP15 per pilotare il LED esterno e il pin GP14 per leggere il pulsante; ora ricostruisci il circuito in modo che il LED e il pulsante siano sulla breadboard nello stesso tempo, connessi a GP15 e GP14 (vedi **Figura 6**). Non dimenticare la resistenza di limitazione della corrente per il LED!

Avvia un nuovo programma in Thonny e inizia importando le due librerie di cui il programma avrà bisogno:

```
import machine
import utime
```

Quindi, imposta entrambi i pin di input e output:

```
led_external = machine.Pin(15, machine.Pin.OUT)
button = machine.Pin(14, machine.Pin.IN,
machine.Pin.PULL_DOWN)
```

Quindi crea un ciclo che legge il pulsante:

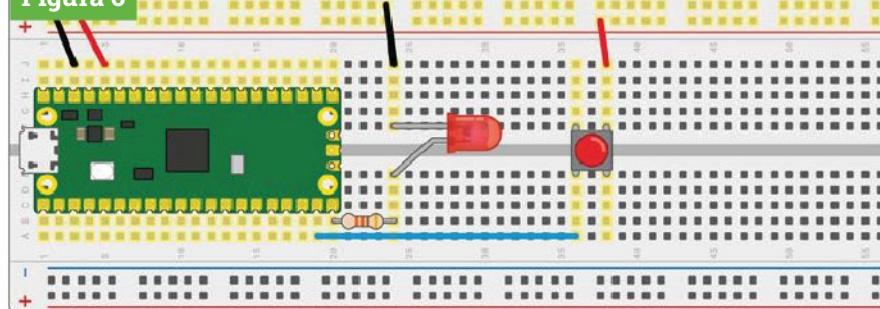
```
while True:
    if button.value() == 1:
```

Invece di visualizzare un messaggio nella Shell, tuttavia, questa volta attiverai il pin di uscita e il LED ad esso collegato in base al valore del pin di ingresso. Digita quanto segue, ricordando che dovrà essere rientrato di otto spazi, che Thonny dovrebbe avere automaticamente gestito quando hai premuto INVIO alla fine della riga sopra:

```
        led_external.value(1)
        utime.sleep(2)
```

È sufficiente per accendere il LED, ma dovrai anche spegnerlo di nuovo quando il pulsante non è premuto. Aggiungi la seguente nuova riga, usando il tasto **BACKSPACE** per eliminare quattro degli otto spazi - il che significa che la linea non farà parte dell'istruzione if, ma farà parte del ciclo infinito:

```
    led_external.value(0)
```

Il programma finito dovrebbe assomigliare a questo:

SUGGERIMENTO

LA RESISTENZA È NASCOSTA

A differenza di un LED, un pulsante non necessita di una resistenza per limitare la corrente. Tuttavia, ha bisogno comunque di una resistenza: quella che è nota come resistenza di pull-up o pull-down, a seconda di come funziona il circuito. Senza una resistenza di pull-up o pull-down, un ingresso rimane flottante - il che significa che ha un segnale "rumoroso", che può attivarsi anche quando non stai premendo il pulsante. Allora dov'è la resistenza in questo circuito? Nascosta nel Pico. Proprio come ha un LED a bordo, il Pico include a bordo una resistenza programmabile collegata a ciascun pin GPIO. Possono essere impostate in MicroPython come resistenze di pull-down o di pull-up, come richiesto dal circuito. Qual è la differenza? Una rersistenza di pull-down collega il pin alla massa, ovvero quando il pulsante non è premuto, l'ingresso sarà 0. Una resistenza di pull-up collega il pin a 3V3, ovvero quando il pulsante non viene premuto, l'ingresso sarà 1. Tutti i circuiti in questa guida utilizzano le resistenze programmabili in modalità pull-down.

Figura 6 Il circuito finito, con un pulsante e un LED

```
import machine
import utime

led_external = machine.Pin(15, machine.Pin.OUT)
button = machine.Pin(14, machine.Pin.IN,
machine.Pin.PULL_DOWN)

while True:
    if button.value() == 1:
        led_external.value(1)
        utime.sleep(2)
    led_external.value(0)
```

Fai clic su Esegui e salva il programma come **Switch.py** sul tuo Pico. All'inizio non accadrà nulla; premi però il pulsante e vedrai il LED accendersi. Rilascia il pulsante; dopo due secondi, il LED si spegnerà finché non premi di nuovo il pulsante.

Congratulazioni: hai costruito il tuo primo circuito che controlla un pin in base all'input di un altro - un mattone per cose più grandi!